

Uniresolution representations of white-matter data from CoCoMac

Raghavendra Singh

IBM Research India, India
ISID Campus, Vasant Kunj, Delhi
raghavs@in.ibm.com

Abstract. Tracing data as collated by CoCoMac, a seminal neuroinformatics database, is at multiple resolutions – white matter tracts were studied for areas and their subdivisions by different reports. Network theoretic analysis of this multi-resolution data often assumes that the data at various resolutions is equivalent, which may not be correct. In this paper we propose three methods to resolve the multi-resolution issue such that the resultant networks have connectivity data at only one resolution. The different resultant networks are compared in terms of their network analysis metrics and degree distributions.

Keywords: White matter tracer data, network theoretic analysis

1 Introduction

Tracing studies have been the conventional mechanism for studying white matter tracts in the primate brains. There have been many studies over the past century, and neuroinformatics databases such as Collation of Connectivity data on the Macaque (CoCoMac) [7] have made a valiant effort to collate these studies. These collations, and networks extracted from the collations, have been very useful in network theoretic analysis of white matter data, providing a number of remarkable insights into the functioning of the brain including distributed and hierarchical structure of cortex [4], topological organization of the cortex [20], functional small-world characteristics, optimal set analysis, and multidimensional scaling [16], small-world characteristics [15], nonoptimal component placement for wire-length [5], structural and functional motifs [14], hub identification and classification [13], exponential degree distribution, and tightly integrated core subnetwork [9].

A core issue with CoCoMac connectivity data is that it is at various resolutions. For example while Yeterian and Pandya have studied white matter tracts of area V4 [19], Lewis and Van Essen have studied tracts of subdivisions of V4 [8]; while Walker [18] studied area 24 of the cingulate cortex and its tracts to prefrontal cortex, Vogt et.al. [17] have studied subareas of 24; similar examples abound in CoCoMac data. Analysis which rely on this data either use subsets of the data, for example [13], or in the case of more comprehensive dataset [9], incorporate all resolutions of the data.

Studying multiple resolutions of a data raises the concern that are all the resolution equivalent? For example with the connectivity network in [9], there is an accompanying hierarchy that codifies the resolutions – areas are organised to represent the recursive subdivisions. The connectivity data is not only reported for all leaf vertices but also for some internal vertices of the hierarchy. So for this dataset would a connection at an internal vertex be equivalent to a connection at its descendant leaf vertex? If not would the network theoretic analysis results change?

In this paper we are proposing methods to resolve the resolution issue in the network of [9], such that the resultant networks have connectivity data at a single resolution. In other words realise a “uni” resolution network that has connectivity only at the leaf vertices, not at the internal vertices of a hierarchy. The central problem in realising such networks is to assign the connectivity data of an internal vertex to which one (or more) of its descendant leaf vertices?

A simple principle would be of inheritance, that is to assign the connectivity data of internal vertex to each of its descendant leaves. This creates a network with many edges, and with the highest resolution possible. The caveat is that some of the connectivity assigned to the leaves may not exist if traced in the Macaque brain. On the other hand the principle of disinheritance could also be applied, internal vertices that have connectivity acquire connectivity data from their descendant vertices. This will create a network with few edges and vertices, but all edges could be traced in primate brain if studied at the right resolution.

A third more interesting principle is based on Kron reduction [3] to remove the internal vertices and reassign its connectivity to leaf vertices. This is followed by sparsification based on electrical resistance [12]. Intuitively Kron reduction preserves paths in the graph, while removing some of the vertices. It has a number of interesting properties, including preserving resistance distance [3]. Resistance distance across an edge is the potential distance induced across it when a current is injected at one end of the edge and extracted at the other end. Spielman and Srivastava have proposed a sparsification algorithm that selects edges based on their resistances and produces high quality spectral sparsifiers of weighted graphs [12]. In this paper we use Kron and sampling to select for each connection of an internal vertex, the leaf vertex that inherits it. Though the resulting connectivity may not be traced in Macaque brain, this principle realises networks that are high resolution and sparse. We compare these three principles in terms of known network theoretic metrics.

2 Methods and Dataset

In this section we outline the algorithms and discuss the dataset used. Let us start by establishing the notation, a network is represented as $G = [V, E, W]$, which is a set of vertices $V = \{v\}$, a set of edges $E = \{e_{uv}\}$ where e_{uv} is a directed edge from vertices u to v , and a set of weights associated with the edges $W = \{w_e\}$. For the case of unweighted networks $w_e = 1 \forall e \in E$, or in short notation $W = \mathbf{1}$. To distinguish between networks we use the superscript

notation, and capital letters to denote sets unless otherwise stated. E_v represent all edges from v , and $E_{:v}$ represent all edges to v . p_{uv} represents the shortest path between u and v in G . $E_{S,T}$ represents all the edges between the vertices in set S and set T .

An input to our algorithms is the connectivity network $G = [V^G, E^G, W^G]$ where the vertices could denote the brain areas as in [9], or could represent people in an organization. An edge e_{uv}^G could imply the presence of a white matter pathway between brain areas as in [9], or a communication channel between people in an organization. The weights would imply the relative strength of the pathway or of the communication channel¹. The other required input is a hierarchy $T = [V^G, E^T, \mathbf{1}]$. This is a tree graph with no cycles, and a root vertex. The hierarchy could represent the subdivisions of the brain into structurally and functionally consistent brain areas as in [9], or could be the org chart in an organization. Thus the inputs of the algorithms are a network G that has edges at internal vertices of a hierarchy T , while the outputs are a network $R = [V^R, E^R, W^R]$ and a hierarchy $M = [V^R, E^M, \mathbf{1}]$ such that R does not have edges at the internal vertices of M .

The algorithm based on the principle of inheritance is outlined in Algo.1. The algorithm finds all internal vertices of G that have connectivity, I (in the algorithm), for each of these vertices it finds descendant leaf nodes, L , and transfers its connectivity to each of the leaf nodes. The output network R has the same number of vertices as G , and a substantially higher number of edges. Output hierarchy M is the same as input hierarchy T .

Algorithm 1 $(R, M) = \text{Inherit}(G, T)$

```

 $R = G;$ 
 $M = T$ 
 $I = \{v : v \in V^G \wedge (\exists e_{v:}^G \vee \exists e_{:v}^G) \wedge e_v^T\}$ 
for  $u \in I$  do
   $L = \{v \in V^G : \exists p_{uv}^T \wedge \nexists e^T(v :)\};$ 
  for  $v \in L$  do
     $W_{v:}^R = W_{v:}^R + W_{u:}^G$ 
     $W_{:v}^R = W_{:v}^R + W_{:u}^G$ 
  end for
   $W_{u:}^R = 0;$ 
   $W_{:u}^R = 0;$ 
   $E_{u:}^R = \emptyset;$ 
   $E_{:u}^R = \emptyset;$ 
end for

```

In Algo.2 the disinheritance principle is used to realise a network with uniresolution connectivity. The algorithm finds all the internal vertices of G that have connectivity I , and for each of these vertices it finds descendants leaf nodes L ,

¹ the network in [9] is unweighted

and transfers the connectivity of the leaf nodes to itself. It then removes the leaf nodes from the output network R and the hierarchy M . The output network R has less number of vertices as G , and a substantially lower number of edges.

Algorithm 2 $(R, M) = \text{DisInherit}(G, T)$

```

 $R = G;$ 
 $M = TG$ 
 $I = \{v : v \in V^G \wedge (\exists e_{v:}^G \vee \exists e_{:,v}^G) \wedge e_{v:}^T\}$ 
for  $u \in I$  do
   $L = \{v \in V^G : \exists p_{uv}^T \wedge \nexists e_{v:}^T \wedge (\exists e_{v:}^G \vee \exists e_{:,v}^G)\};$ 
  for  $v \in L$  do
     $W_{u:}^R = W_{u:}^R + W_{v:}^G$ 
     $W_{:,u}^R = W_{:,u}^R + W_{:,v}^G$ 
     $W_{u:}^G = 0; W_{:,u}^G = 0;$ 
     $E_{u:}^R = \emptyset; E_{:,u}^R = \emptyset;$ 
     $E_{u:}^G = \emptyset; E_{:,u}^G = \emptyset;$ 
  end for
end for
for  $u \in V$  do
   $C = \{v \in V^R : \exists p_{uv}^T\};$ 
  for  $v \in C$  do
     $V^R = V^R \setminus v; V^m = V^m \setminus v;$ 
  end for
   $E_{u:}^m = \emptyset$ 
end for

```

The above two algorithms are simple and in either case follow a take-all principle – either all the descendants inherit equally from their ancestors, or an ancestor disinherits all its descendants. The resultant networks are clearly the opposite end of a spectrum, and there are many networks in the entire range of the spectrum. In the last algorithm we propose that each edge of the original network should be represented by *one and only one* edge in the resultant network. Further the network should have the highest resolution possible, in other words the resultant edges should be at the leaf nodes of a hierarchy M , with $M == T$. Towards this we propose the KronSampling algorithm in Algo.3. The intuition behind this algorithm is to find for each edge at an internal vertex in G , an edge at its descendant vertices. To select this winner-take-all edge we use precise ordering of edges based on depth from the root in the hierarchy, and probability based on effective resistance and number of times the edge was reported in the sub-hierarchy. This allows us to realise a network with sparsification guarantees [12].

The algorithm requires three modules, each of which are described next. The Kron reduction of a graph is again a graph whose Laplacian matrix is obtained by the Schur complement of the original Laplacian matrix with respect to a subset of nodes [3]. It is defined as follows, let \mathcal{L} be the Laplacian of G [2] – if we consider W as a matrix where the uv element is set to w_{uv} and other

elements to zero, and D is a diagonal matrix with elements $d_{uu} = \sum_v w_{uv}$, then the Laplacian matrix is $\mathcal{L} = D - W$. Now let U be a subset of V , the Kron reduction of the Laplacian can be the given by,

$$\mathcal{L}^K := \mathcal{L}_{UU} - \mathcal{L}_{UU^c} \mathcal{L}_{U^c U^c}^{-1} \mathcal{L}_{U^c U} \quad (1)$$

where \mathcal{L}_{AB} represents the $|A| \times |B|$ sub matrix consisting of all entries in \mathcal{L} whose row index is in A , and whose column index is in B . We can then uniquely associate with Laplacian \mathcal{L}^K a reduced weighted graph $K = [V^K = U, E^K, W^K]$ by letting,

$$W_{uv}^K = -\mathcal{L}_{uv}^K; \text{ if } u \neq v \quad (2)$$

$$= 0 \text{ else} \quad (3)$$

The edges in E^K exist wherever W^K is nonzero. U^c represents the compliment of U in V . The many properties of Kron reduction are detailed in [3] and references therein. The module $K = \text{Kron}(G, U)$ takes the graph G , finds its Laplacian \mathcal{L} , and uses equation(1) and then equations(2, 3) to output the reduced network K .

The second module $R = \text{Resistance}(K)$ outputs a network $R = [V^R = V^K, E^R = E^K, W^R]$ whose weights are the effective resistance of the edges of input network $K = [V^K, E^K, W^K]$. If \mathcal{L} is the Laplacian of K then effective resistance is defined by,

$$W_{uv}^R = \mathcal{L}_{uu}^\dagger + \mathcal{L}_{vv}^\dagger - 2\mathcal{L}_{uv}^\dagger \quad (4)$$

where \mathcal{L}^\dagger is the Moore-Penrose inverse of \mathcal{L} . Again [3] and furthermore [12] have an excellent discussion on the properties of the effective resistance of a network.

The final module $P = \text{Probability}(O, Z)$ takes networks O and Z that have the same vertices, the same edges, but different weights on edges, and outputs a network P with same vertices and edges, and weights defined by,

$$W_{uv}^P = \frac{W_{uv}^O \times W_{uv}^Z}{\sum_{u,v} (W_{uv}^O \times W_{uv}^Z)} \quad (5)$$

Clearly each weight assigns a probability distribution over the edges in the network P . Note that if G was unweighted, i.e., $w_{uv}^g = 1 \forall u, v$ as in the network of [9], the weights of the resultant network R for Inherit algorithm will reflect the number of times connectivity is reported in a sub-hierarchy of G , i.e., $w_{u,v}^R = k$ implies that there are k edges between the descendants of u and v in G ². Thus W^R can be taken atleast as an indicator of the number of times tracer studies were done for the incident areas, albeit at different resolutions. We will use W^R along with the effective resistance to sample edges.

Having estimated the probability from both the effective resistance and count of times connectivity is reported in a sub-hierarchy, we sort all edges of original

² if G is positive weighted this interpretation will still hold, with $w_{uv}^R \propto k$

network in increasing order of product of their depth first distance from root vertex Br . Thus the first edge to be considered is one which is between the farthest leaf vertices of G , and all leaf-leaf edges are added to E^R . Then among the remaining edges of G the most likely edges are sampled if they do not already exist in R network.

Algorithm 3 $(R, M) = \text{KronSampling}(G, T)$

```

 $R = G;$ 
 $M = T;$ 
 $L = \{v \in V^g : \nexists e_{v:}^T \wedge (\exists e_{v:}^G \vee \exists e_{:v}^G)\};$ 
 $K = \text{Kron}(G, L);$ 
 $O = \text{Resistance}(K);$ 
 $Z = \text{Inherit}(G, T);$ 
 $P = \text{Probability}(O, Z);$ 
 $D = \text{DepthFirstSearch}(M, 1);$ 
 $E^d = \text{Sort}(E^G, D);$ 
 $W^R = 0; V^R = V^G; E^R = \emptyset;$ 
for  $e_{uv} \in E^d$  do
   $L_u = \{w \in V^G : \exists p_{uw}^T \wedge \nexists e_{w:}^T \wedge (\exists e_{w:}^G \vee \exists e_{:w}^G)\};$ 
   $L_v = \{w \in V^G : \exists p_{vw}^T \wedge \nexists e_{w:}^T \wedge (\exists e_{w:}^G \vee \exists e_{:w}^G)\};$ 
   $L_u = L_u \cup u;$ 
   $L_v = L_v \cup v;$ 
  if  $\nexists e \in E_{L_u L_v}^R$  then
     $(s, t) = \arg \max W_{L_u L_v}^P;$ 
     $W_{st}^R = 1;$ 
     $E^R = E^R \cup e_{st}^R;$ 
  end if
end for

```

We study the long range network of the Macaque brain as derived by Modha and Singh [9]. The network is based on tracing studies of the Macaque brain compiled by the online database CoCoMac [7]. It covers 383 cortical and sub-cortical brain areas and codes the presence of 6602 directed projections between these areas. The brain areas are arranged in a hierarchal brain map, which is consistent with a recursive parcellation of the brain [9]. 351 of the 383 areas have connectivity; the remaining areas are container or super-areas that hold the hierarchy together.³ The hierarchal map divides the brain (Br) into basal ganglia (BG), diencephalon (DiE), and cortex (Cx). Cortex is divided into 6 lobes, temporal (TL#2), occipital (OC#2), parietal (Pl#6), frontal (FL#2), cingulate gyrus (CgG#2) and insula (Ins). These super-areas are further sub-divided into other super-areas and brain areas.

³ We differentiate a super-area from a brain area in that a super-area is sub-divided into brain areas and it does not report any projections.

3 Results and Discussion

In this section we present the spy plots for all four networks in Fig. 1(top), while (bottom) four plots show the degree distribution of the networks in semi-log scale fitted to a maximum entropy exponential distribution. In a network, degree of a vertex is the total number of edges that it touches. The tail behavior of the frequency distribution of degrees is a key signature of how connectivity is spread among vertices [9]. In Tables 2 –5 the top areas for various network theoretic metrics are shown for each network. Besides all other observations we must emphasise that Table 5 shows that the prefrontal cortex predominates the Top-10 table, which is consistent with results in [9]. The table was computed using Pajek [11]. Due to lack of space we will conclude that we have provided three methods for taking a multi-resolution network and converting it to a single-resolution. In terms of network theoretic analysis we see that the KronSampling network is most similar to the original network. We must point out that these algorithms are novel and each of them can be used to solve similar multi-resolution problems.

Metric	Original Network	Inherit	DisInherit	KronSampling
#Vertices	383	383	66	383
#Edges	6491	22236	793	4663
Density	0.0509	0.3143	0.182	0.0659
Reciprocity	0.4223	0.3567	0.7238	0.4160
Diameter	6	6	4	6
Char. path lengt	2.614	1.932	1.771	2.4836
Mean Clustering Coefficient (Directed)	0.3140	0.5926	0.5066	0.2298

Table 1. Metric for four different networks. The number of vertices reported are the number in the hierarchy, some of them may not have connectivity. Original Network is from [9]. The various metrics are also described in the same paper.

References

1. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems 30(1-7), 107–117 (April 1998), <http://portal.acm.org/citation.cfm?id=297810.297827>
2. Chung, F.: Spectral graph theory, vol. 92. Amer Mathematical Society (1997)
3. Dorfler, F., Bullo, F.: Kron reduction of graphs with applications to electrical networks. Circuits and Systems I: Regular Papers, IEEE Transactions on 60(1), 150–163 (2013)
4. Felleman, D.J., Van Essen, D.C.: Distributed hierarchical processing in primate cerebral cortex. Cereb. Cortex 1, 1–47 (1991)
5. Kaiser, M., Hilgetag, C.C.: Nonoptimal component placement, but short processing paths, due to long-distance projections in neural systems. PLoS Comput. Biol. 2(7, e95), 805–815 (2006)

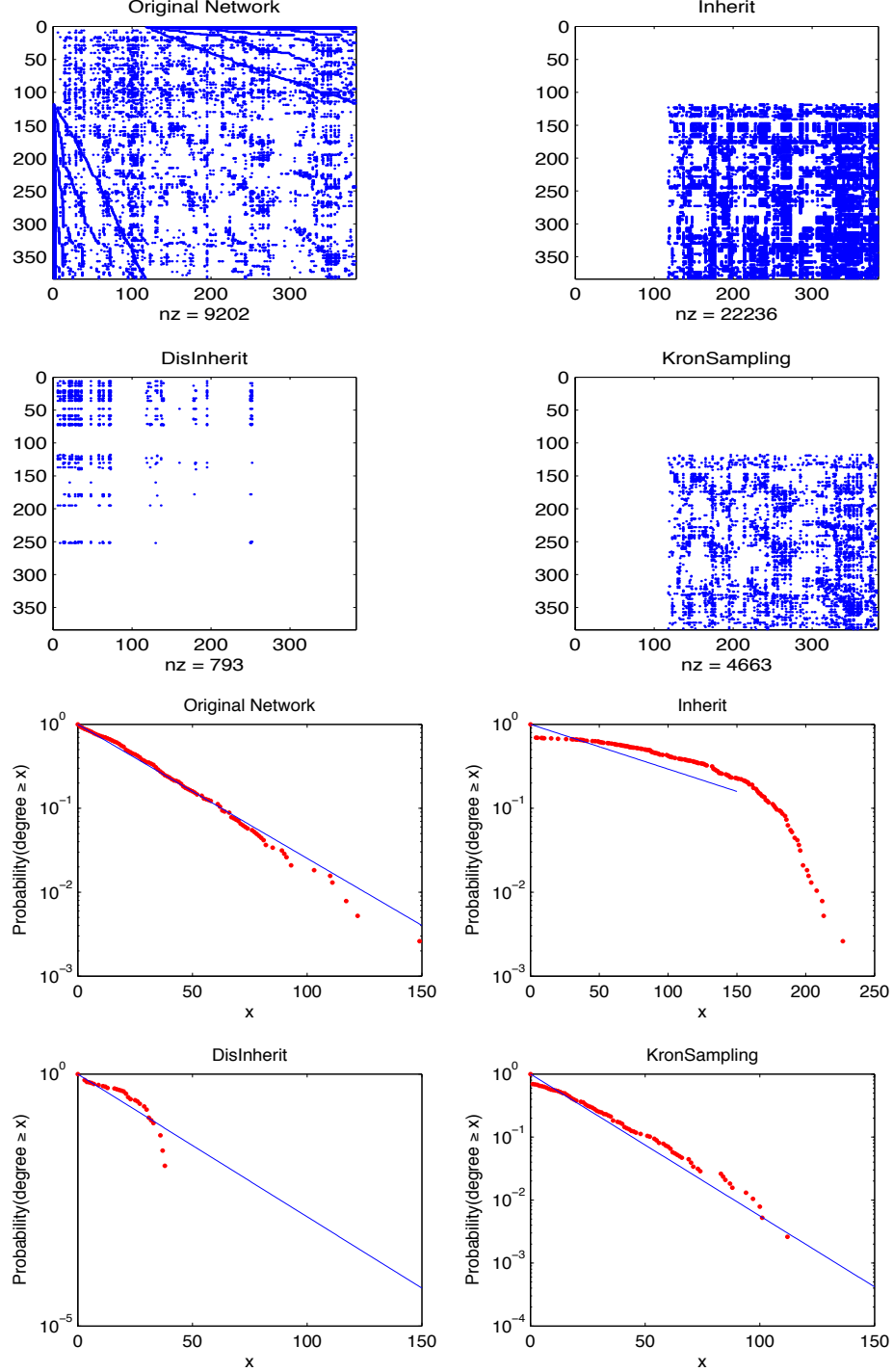


Fig. 1. The top 4 are spy plots of the adjacency matrix of each network. The matrices have been reordered such that the internal vertices of the original hierarchy of [9] are ordered first, while leaf vertices are ordered last. Thus both Inherit and KronSampling networks have connectivity only between leaf vertices. Disinherit network, as explained, has connectivity between internal vertices, but the descendants of these vertices can be removed thus again ensuring that connectivity is between leaf vertices of resultant hierarchy. The bottom 4 plots shows the complementary cumulative distribution of the maximum entropy exponential distribution fit over the range of the data for each network. These plots suggest that the hypothesis of the maximum entropy exponential distribution is consistent with the data particularly for Original and Kron Sampling networks.

<i>Characteristic</i>	<i>Rank</i> →	1	2	3	4	5	6	7	8	9	10
Integrator	In-Degree	32	46	12o	12l	11	24	F7	14	8A	LIP
	In-Closeness	46	12o	32,11	24	12l	MD	8A	23c	8B	LIP,F7
	Authorities	32	12o	46	11	12l	24	14	F7	MD	9
Distributor	Out-Degree	46	24	TF	9	13	13a	TH	TE,LIP	PGm	V2
	Out-Closeness	46	24	TF	TE	9	TH	LIP	PGm	23,PM#3,45	12
	Hubs	46	24	9	TF	TE	TH	13	32	23	PM#3
Intermediary	Betweenness	24	46	LIP	13a	MD	32	TF	PIT	13	PS
	PageRank	32	MD	46	36r	PIT	12o	24	23c	12l	11

Table 2. Top ten brain areas according to several metrics of topological centrality for the directed version of the Original network [9]. The cells are color coded according to the following scheme. If the area is a sub-area of pre frontal cortex (Pfc) in [9] then its coloured red, else if sub-area of motor, parietal lobe, or Insular Cortex (6#1, M1, Pl#6 or Insula) its coloured pink. If area is a sub-area of temporal or occipital lobe (TL#2 or OC#2) then its coloured green, and remaining limbic areas (CgG#2, Tha, BG) are coloured blue. Cells that have multiple areas are not coloured.

<i>Characteristic</i>	<i>Rank</i> →	1	2	3	4	5	6	7	8	9	10
Integrator	In-Degree	F7	24c	24b	24d	24a	TPag	11m	TPg	12o	46v
	In-Closeness	TPdgv	TPdgd	TPag	TPg	36p	12o	12l	11l	11m	46v
	Authorities	24b	24c	24a	24d	TPag	12o	TFL	TPg	TFM	F7
Distributor	Out-Degree	24c	24b	24d	24a	PFG#1	PF#1	Idg	Iam	PS	Ial
	Out-Closeness	PFG#1	PF#1	24c	24d	24b	24a	TPdgv	TPdgd	TPag	TPg
	Hubs	24c	24b	24d	24a	PS	CITv	Iam	Idg	45A	45B
Intermediary	Betweenness	24c	24b	24d	24a	PF#1	13a	PFG#1	PS	F7	F2
	PageRank	MDpm	MDfi	24c	24b	MDpc	24d	MDmf	MDdc	MDcd	24a

Table 3. Top ten brain areas according to several metrics of topological centrality for the directed version of Inherit network. Colour coding as in Table 2.

<i>Characteristic</i>	<i>Rank</i> →	1	2	3	4	5	6	7	8	9	10
Integrator	In-Degree	FD#1	8	6#1	7#1	Tha	23	24	STS	PFCorb	Per#1
	In-Closeness	FD#1	Tha	8	6#1	7#1	23	24	STS	PFCorb	Per#1
	Authorities	# FD#1	8	7#1	23	Tha	6#1	PHC	24	STS	Per#1
Distributor	Out-Degree	6#1	FD#1	7#1	Insula	8	24	OA	STS	TE	Tha
	Out-Closeness	6#1	FD#1	7#1	Insula	8	OA	STS	24	TE	Tha
	Hubs	FD#1	7#1	Insula	6#1	24	STS	8	Tha	PFCorb	TE
Intermediary	Betweenness	6#1	Tha	8	FD#1	OA	TE	7#1	Amyg	V2	STS
	PageRank	Tha	6#1	8	Cd	FD#1	23	7#1	TE	STS	Amyg

Table 4. Top ten brain areas according to several metrics of topological centrality for the directed version of DisInherit network. Colour coding as in Table 2.

Characteristic	Rank \rightarrow	1	2	3	4	5	6	7	8	9	10
Integrator	In-Degree	14r	32	11m	12o	12l	13a	PF#1	10o	TFL	8Ac
	In-Closeness	12l	14r	13a	10o	12o	11m	46v	PF#1	36r	PITv
	Authorities	12o	12l	32	14r	11m	13a	TFL	10o	36r	23c
Distributor	Out-Degree	13a	24a	TFL	46dr	12l	D9	TH	Iai	11m	F5
	Out-Closeness	45B	13a	12l	D9	46dr	PGm	TFL	24a	TH	TPg
	Hubs	13a	12l	TFL	D9	24a	11m	14r	10o	Iai	TH
Intermediary	Betweenness	13a	24a	PF#1	TFL	PITv	12l	LIPi	8Ac	32	V2
	PageRank	MDcd	32	PITv	14r	12l	13a	12o	11m	24a	23c

Table 5. Top ten brain areas according to several metrics of topological centrality for the directed version for KronSampling network. Colour coding as in Table 2.

- Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. J. ACM 46(5), 604–632 (1999)
- Kötter, R.: Online retrieval, processing, and visualization of primate connectivity data from the CoCoMac database. Neuroinformatics 2, 127–144 (2004)
- Lewis, J.W., Van Essen, D.C.: Corticocortical connections of visual, sensorimotor, and multimodal processing areas in the parietal lobe of the macaque monkey. Journal of Comparative Neurology 428(1), 112–137 (2000)
- Modha, D.S., Singh, R.: Network architecture of the long-distance pathways in the macaque brain. Proceedings of the National Academy of Sciences USA 107(30), 13485–13490 (2010), <http://www.pnas.org/content/107/30/13485.abstract>
- Newman, M.E.J.: A measure of betweenness centrality based on random walks. Social Networks 27(1), 39–54 (2005)
- de Nooy, W., Mrvar, A., Batagelj, V.: Exploratory Social Network Analysis with Pajek. Cambridge University Press (2005)
- Spielman, D.A., Srivastava, N.: Graph sparsification by effective resistances. SIAM Journal on Computing 40(6), 1913–1926 (2011)
- Sporns, O., Honey, C.J., Kötter, R.: Identification and classification of hubs in brain networks. PLoS ONE 2(10), e1049 (10 2007), <http://dx.plos.org/10.1371/journal.pone.0001049>
- Sporns, O., Kötter, R.: Motifs in brain networks. PLoS Biol. 2(11), 1910–1918 (2004)
- Sporns, O., Zwi, J.: The small world of the cerebral cortex. Neuroinformatics 2, 145–162 (2004)
- Stephan, K.E., Hilgetag, C.C., Burns, G.A.P.C., O’Neill, M.A., Young, M.P., Kötter, R.: Computational analysis of functional connectivity between areas of primate cerebral cortex. Phil. Trans. R. Soc. Lond. B 355, 111–126 (2000)
- Vogt, B.A., Pandya, D.N., Rosene, D.L.: Cingulate cortex of the rhesus monkey: I. cytoarchitecture and thalamic afferents. Journal of Comparative Neurology 262(2), 256–270 (1987)
- Walker, A.E.: A cytoarchitectural study of the prefrontal area of the macaque monkey. Journal of Comparative Neurology 73(1), 59–86 (1940)
- Yeterian, E., Pandya, D.: Corticostriatal connections of extrastriate visual areas in rhesus monkeys. Journal of Comparative Neurology 352(3), 436–457 (1995)
- Young, M.P.: Objective analysis of the topological organization of the primate cortical visual system. Nature 358, 152–155 (1992)